AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application.

1. (Previously Presented) A method of selecting a heuristic class for data placement in a distributed storage system comprising the steps of:

forming an integer program for each of a plurality of heuristic classes, each of the heuristic classes providing a technique for placing data within the distributed storage system, each of the integer programs comprising an objective of minimizing a replication cost for placing the data;

solving each of the integer programs which provide the replication cost for each of the heuristic classes; and

selecting the heuristic class having a low replication cost.

2. (Previously Presented) A method of selecting a heuristic class for data placement in a distributed storage system comprising the steps of:

forming a general integer program which models placing data within the distributed storage system;

forming a specific integer program which models a heuristic class that provides a technique for placing the data within the distributed storage system, the general and specific integer programs each comprising an objective of minimizing a replication cost for placing the data;

solving the general integer program which provides a general lower bound for the replication cost;

solving the specific integer program which provides a specific lower bound for the replication cost; and

selecting the heuristic class if a difference between the general lower bound and the specific lower bound is within an allowable amount.

3. (Original) The method of claim 2 wherein inputs used in the steps of forming the general and specific integer programs comprise a system configuration, a workload,

and a performance requirement.

4. (Original) The method of claim 3 wherein the performance requirement comprises a bi-modal performance metric.

5. (Original) The method of claim 4 wherein the bi-modal performance metric comprises a criterion and a ratio of successful attempts to total attempts.

6. (Original) The method of claim 3 wherein the performance requirement comprises a data access latency.

7. (Original) The method of claim 3 wherein the performance requirement comprises a data access bandwidth.

8. (Original) The method of claim 3 wherein the performance requirement comprises a data update time.

9. (Original) The method of claim 3 wherein the performance requirement comprises an average data access latency.

10. (Original) The method of claim 3 wherein the performance requirement comprises a data availability requirement.

11. (Original) The method of claim 3 wherein the general integer program comprises general constraints which model the data placement irrespective of the heuristic class for the data placement.

12. (Original) The method of claim 11 wherein the general constraints comprise a performance constraint which models the performance requirement.

13. (Original) The method of claim 11 wherein the specific integer program

comprises the general constraints and a specific constraint.

14. (Original) The method of claim 12 wherein the specific constraint comprises a storage constraint.

15. (Original) The method of claim 12 wherein the specific constraint comprises a replica constraint.

16. (Original) The method of claim 12 wherein the specific constraint comprises a routing knowledge constraint and further wherein the routing knowledge constraint models an extent to which a data storage node knows of replicas of data objects stored on other data storage nodes.

17. (Original) The method of claim 12 wherein the specific constraint comprises an access knowledge constraint and further wherein the access knowledge constraint models an extent to which a data storage knows of access to replicas of data objects by clients accessing other data storage nodes.

18. (Original) The method of claim 12 wherein the specific constraint comprises an activity history constraint.

19. (Original) The method of claim 12 wherein the specific constraint comprises a reactive placement constraint.

20. (Original) The method of claim 3 wherein the system configuration comprises a plurality of data storage nodes coupled by a plurality of network links.

21. (Original) The method of claim 20 wherein the system configuration further comprises a plurality of clients coupled to the data storage nodes.

22. (Original) The method of claim 21 wherein the workload comprises at least some

of the clients requesting data objects stored on the data storage nodes.

23. (Original) The method of claim 22 wherein the workload further comprises at least some of the clients storing some of the data objects on the data storage nodes.

24. (Previously Presented) A method of selecting a heuristic class for data placement in a distributed storage system comprising the steps of:

forming a general integer program which models placing data within the distributed storage system;

forming a plurality of specific integer programs which model a plurality of heuristic classes, each of the heuristic classes providing a technique for placing the data within the distributed storage system, the general and specific integer programs each comprising an objective of minimizing a replication cost for placing the data;

solving the general integer program which provides a lower bound for the replication cost;

solving the specific integer programs which provides the replication cost for each of the heuristic classes; and

selecting a particular heuristic class correlated to a low replication cost if a difference between the lower bound and the low replication cost is within an allowable amount.

25. (Previously Presented) A computer readable memory comprising computer code for implementing a method of selecting a heuristic class for data placement in a distributed storage system, the method of selecting the heuristic class comprising the steps of:

forming an integer program for each of a plurality of heuristic classes, each of the heuristic classes providing a technique for placing the data within the distributed storage system, each of the integer programs comprising an objective of minimizing a replication cost for placing the data;

solving each of the integer programs which provide the replication cost for

each of the heuristic classes; and

selecting the heuristic class having a low replication cost.

26. (Previously Presented) A computer readable memory comprising computer code for implementing a method of selecting a heuristic class for data placement in a distributed storage system, the method of selecting the heuristic class comprising the steps of:

forming a general integer program which models placing data within the distributed storage system;

forming a specific integer program which models a heuristic class that provides a technique for placing the data within the distributed storage system, the general and specific integer programs each comprising an objective of minimizing a replication cost for placing the data;

solving the general integer program which provides a general lower bound for the replication cost;

solving the specific integer program which provides a specific lower bound for the replication cost; and

selecting the heuristic class if a difference between the general lower bound and the specific lower bound is within an allowable amount.

27. (Previously Presented) A computer readable memory comprising computer code for implementing a method of selecting a heuristic class for data placement in a distributed storage system, the method of selecting the heuristic class comprising the steps of:

forming a general integer program which models placing the data within the distributed storage system;

forming a plurality of specific integer programs which model a plurality of heuristic classes, each of the heuristic classes providing a technique for placing the data within the distributed storage system, the general and specific integer programs each comprising an objective of minimizing a replication cost for placing the data;

solving the general integer program which provides a lower bound for the replication cost;

solving the specific integer programs which provides the replication cost for each of the heuristic classes; and

selecting a particular heuristic class correlated to a low replication cost if a difference between the lower bound and the low replication cost is within an allowable amount.